

Figures

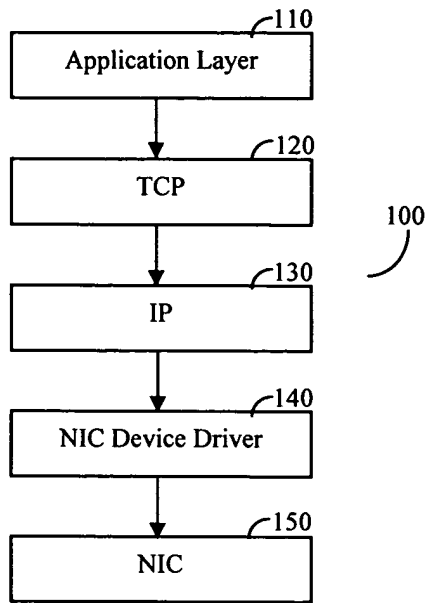


FIGURE 1 (PRIOR ART)

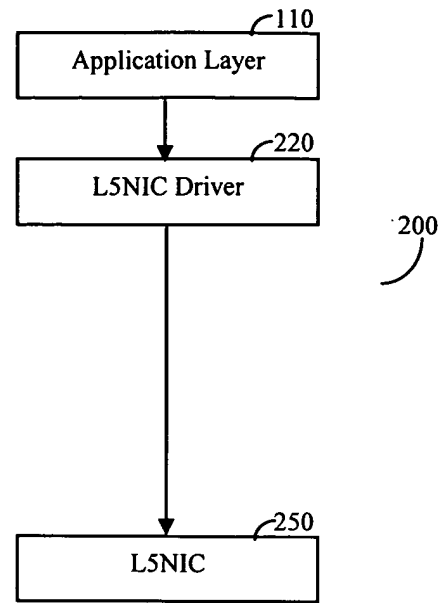


FIGURE 2

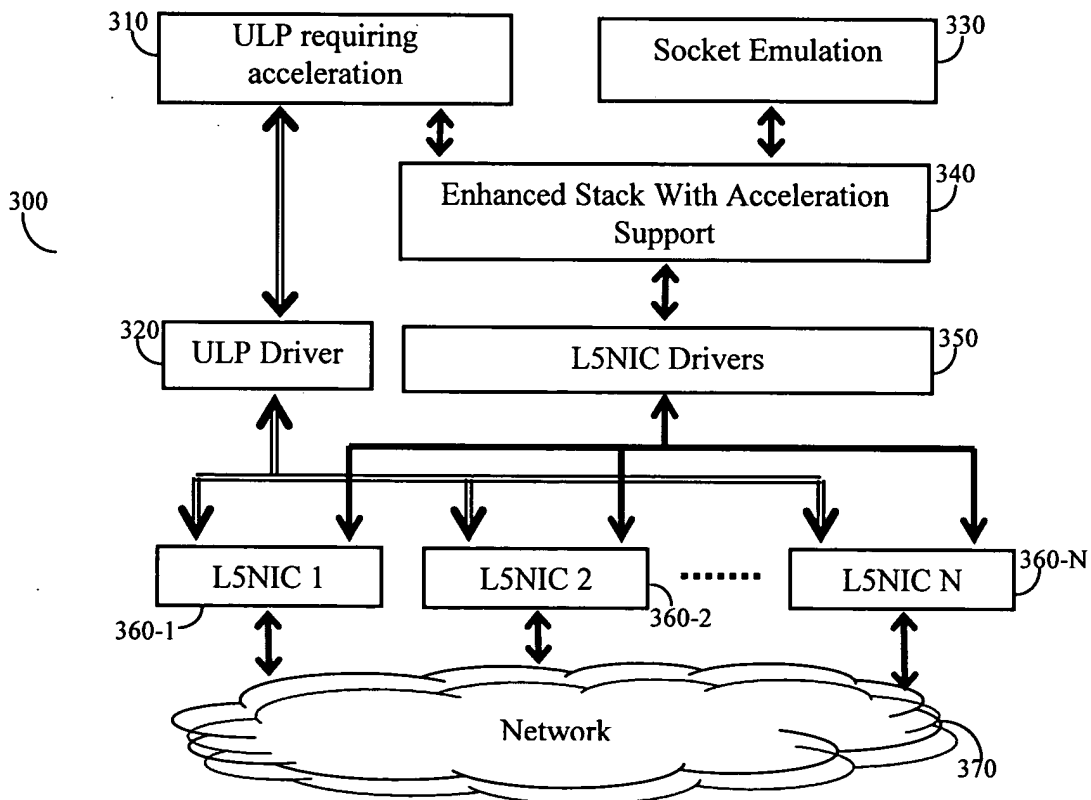


FIGURE 3

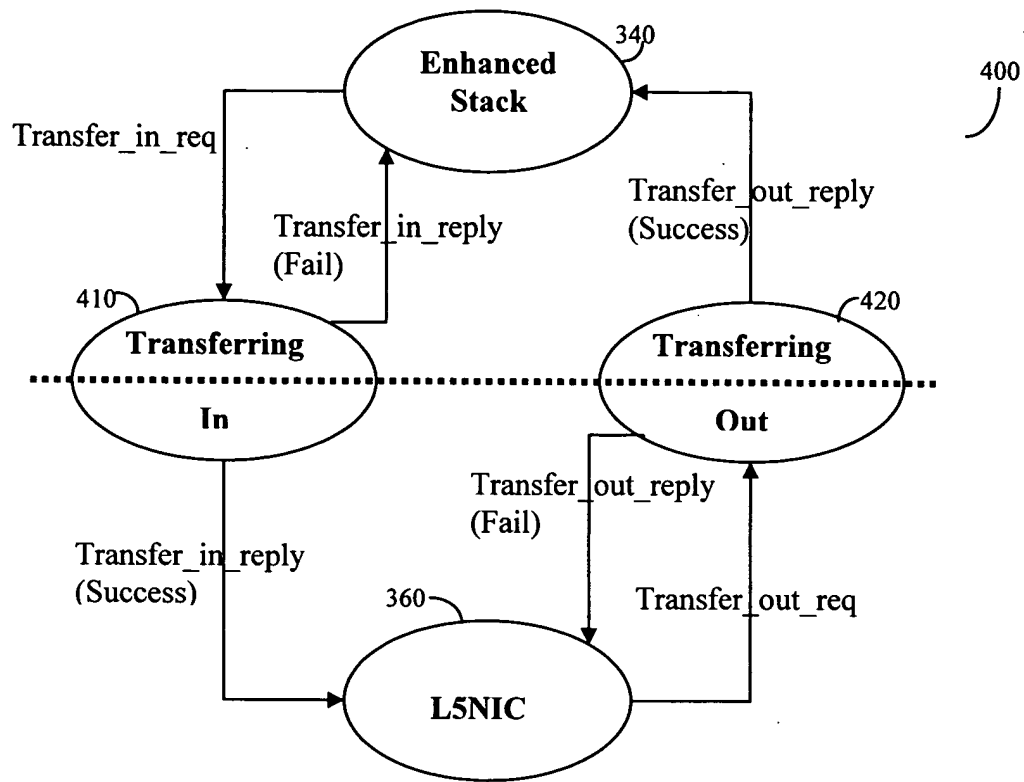


FIGURE 4

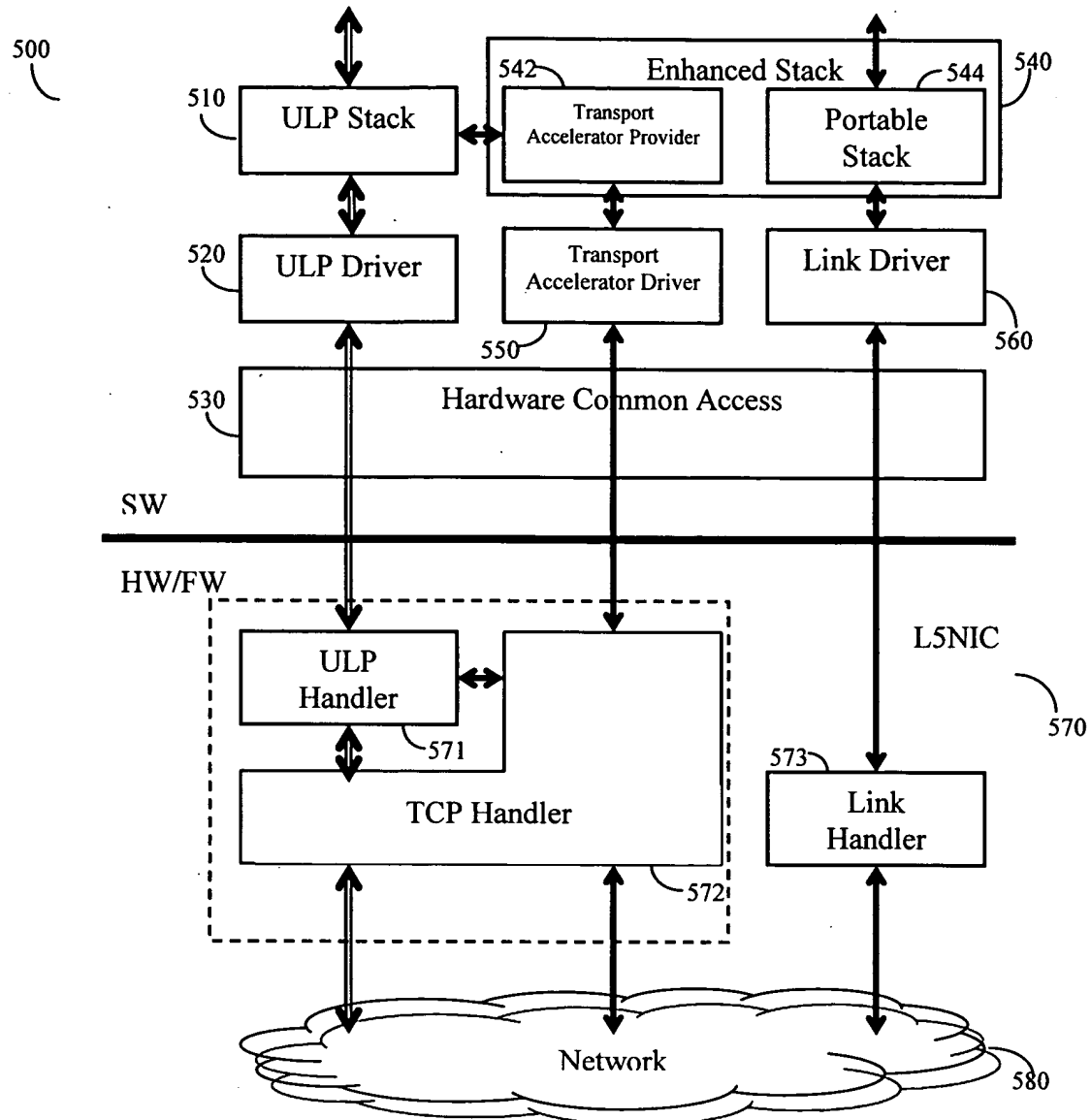


FIGURE 5

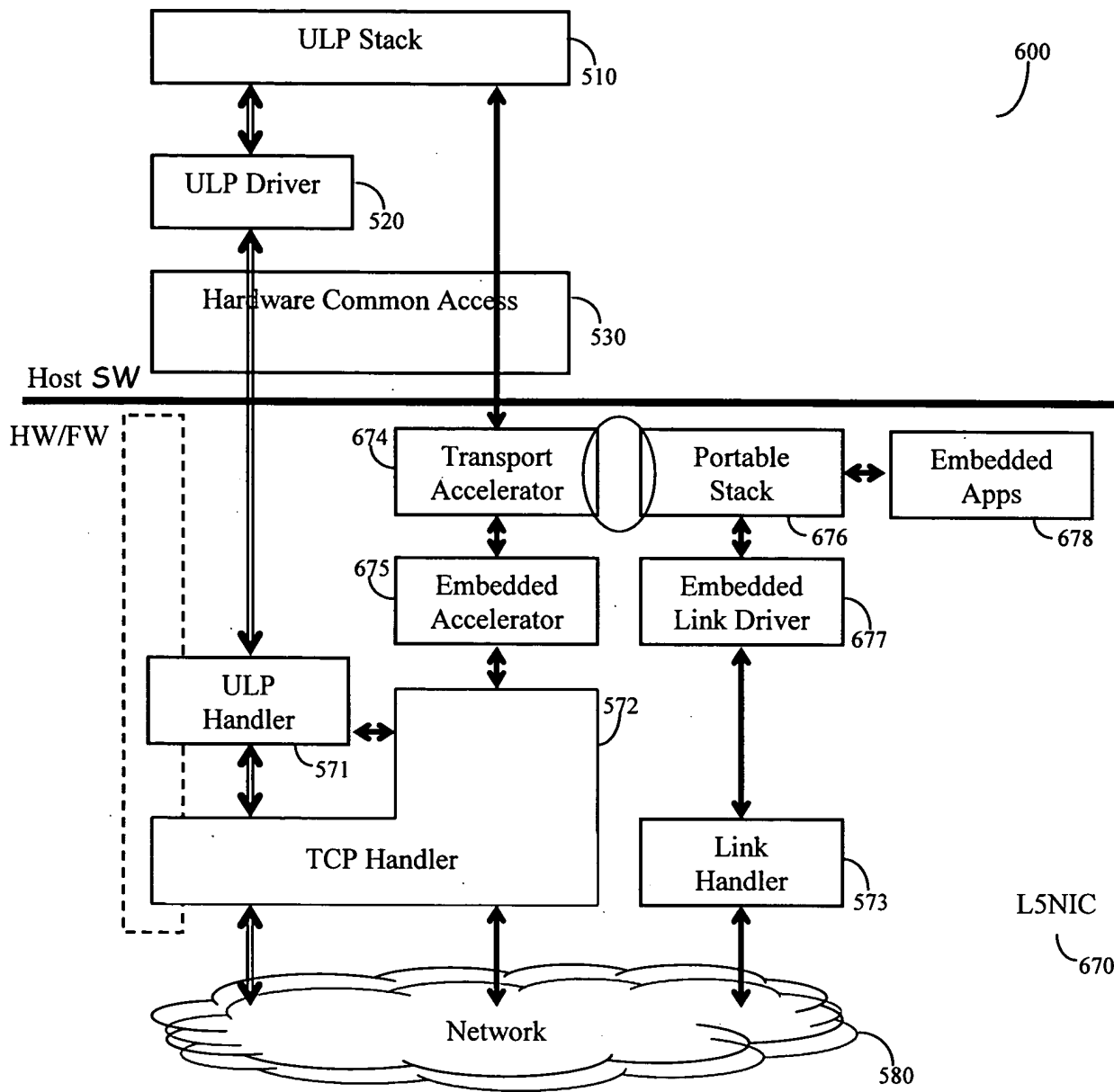


FIGURE 6

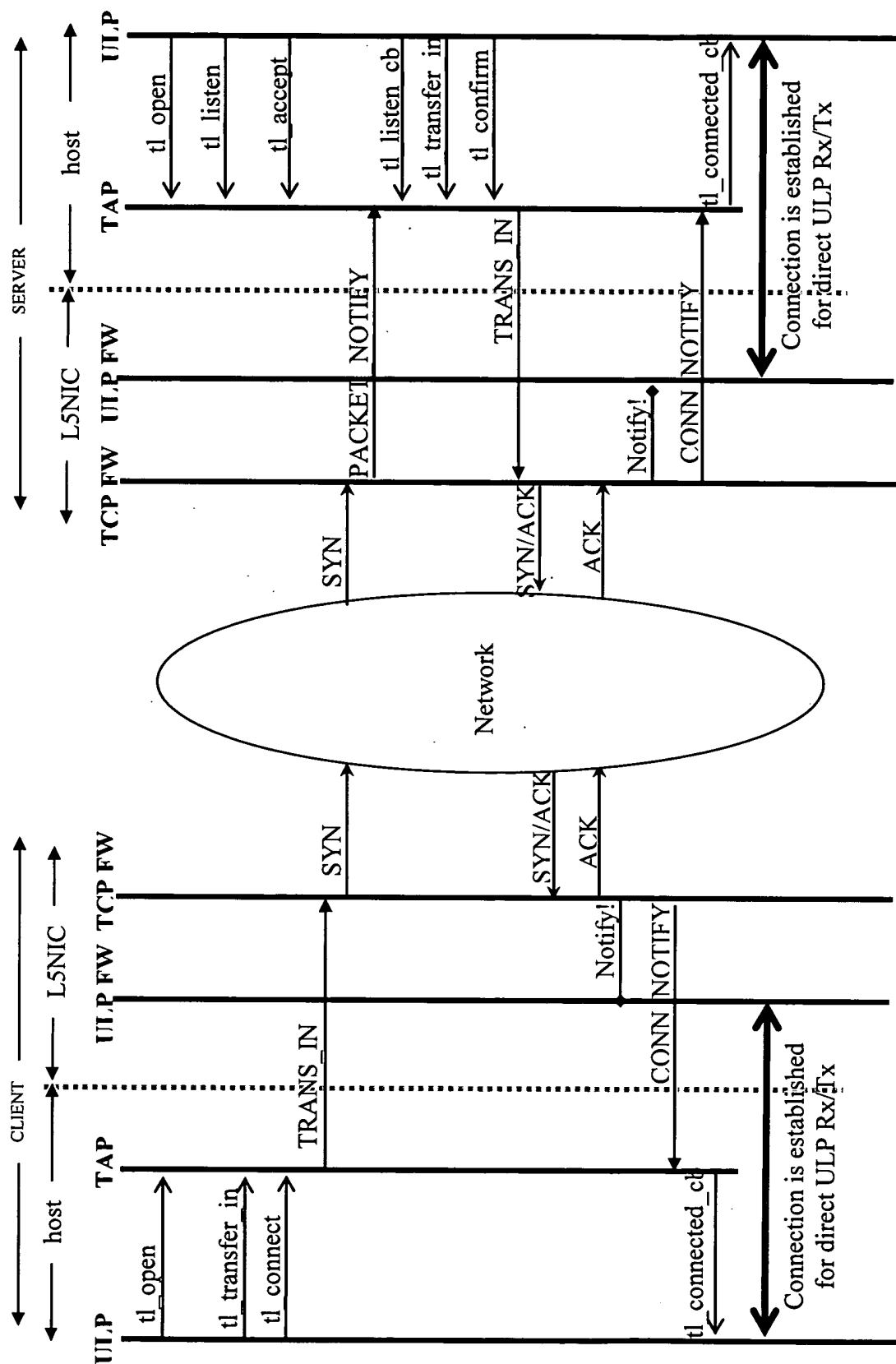


FIGURE 7

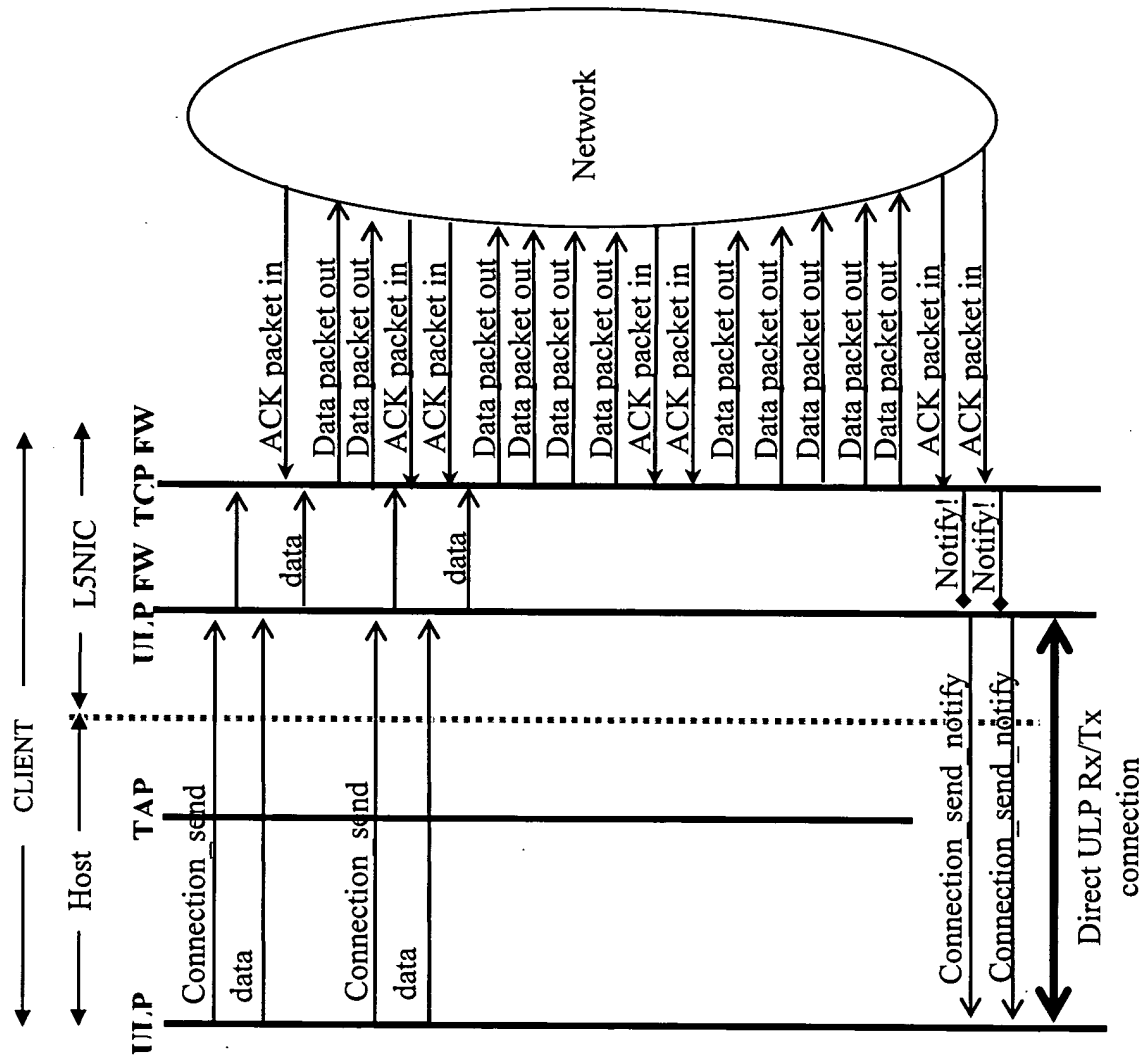


FIGURE 8

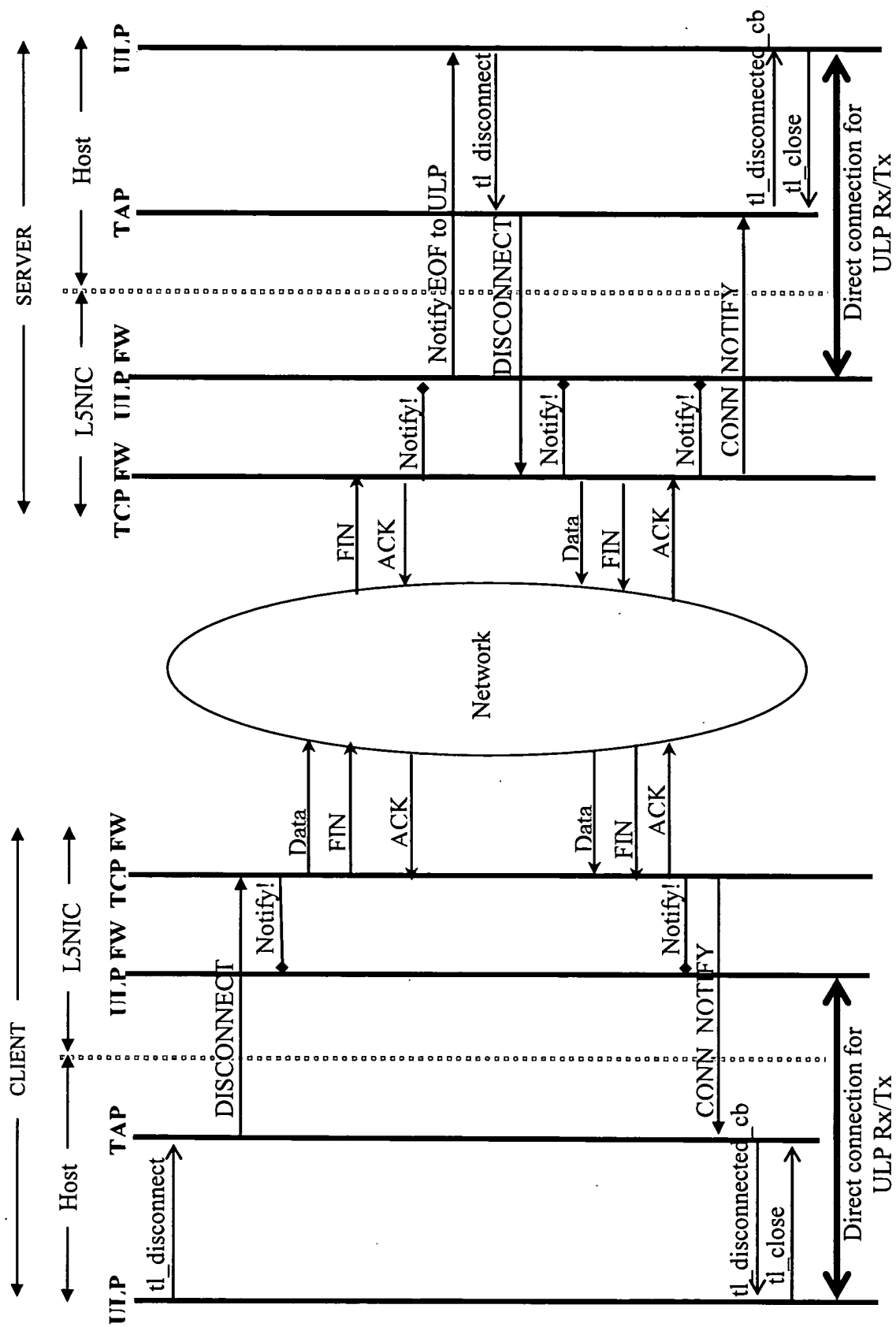


FIGURE 9

CONNECTION_TRANSFER_IN

This command is sent in order to transfer a connection acceleration (processing) into the L5NIC. This command contains all necessary information required to start a new or continue processing of an existing connection. The data associated with this command, in a case of TCP/IP connection, may include, but is not limited to: Connection 4tuple, Initial sequence number, unacknowledged sequence number, acknowledgement sequence number, current sent TimeStamp, current received (echo) timestamp, remote/local negotiated window scale values, and others. A unique identifier for connection reference is used, recognizable by the L5NIC and by a host. This command may be sent before the connection is actually initialized and then the connection is fully offloaded, i.e., even SYN exchange for establishing a TCP connection will be executed on the L5NIC.

CONNECTION_TRANSFER_OUT

Used when it is necessary to transfers a connection out of the L5NIC. The only required data in this message is the connection reference.

CONNECTION_DISCONNECT

A command used in order to gracefully close (FIN exchange) an offloaded (accelerated) connection. The only required data in this message is the connection reference.

CONNECTION_ABORT

Used in order to abort an offloaded (accelerated) connection. The only required data in this message is the connection reference.

CONNECTION_SHUTDOWN_TX

This command is used in order to gracefully close the write side of a connection (send FIN). The only required data in this message is the connection reference.

CONNECTION_SEND

Used for the purpose of transmitting data over a connection. The data associated with this command, in a case of TCP/IP connection, may include, but is not limited to: connection reference, list of buffers in the host memory and their length. These buffers contain data to be transferred over the connection.

FIGURE 10 A

CONNECTION_RECEIVE

Used to request data reception from a connection into connection specific receive buffers in a host memory. The data associated with this command, in a case of TCP/IP connection may include, but is not limited to: connection reference, list of connection specific buffers in the host memory and their length. These buffers will be filled by the L5NIC with data received over the connection.

CONNECTION_MANAGE

This command is used in order to change certain connection parameters. The data associated with this command, in a case of TCP/IP connection may include, but is not limited to: connection reference, receive window size to advertise, routing related information, disable/enable nagle algorithm.

CONNECTION_QUERY

A request sent in order to query certain connection parameters. The data associated with this command, in a case of TCP/IP connection, may include, but is not limited to: connection reference, parameters to be reported.

CONNECTION_QUERY_REPLY

A response from a L5NIC with data requested by the corresponding CONNECTION_QUERY request. The data associated with this reply may include, but is not limited to: connection reference, reported parameters.

CONNECTION_SYNC

A request sent in order to synchronize with a connection. This is an auxiliary message used to flush existing message pipes between a host and a L5NIC. The data associated with this command, in a case of TCP/IP connection, may include, but is not limited to: connection reference.

CONNECTION_SYNC_REPLY

Acknowledges reception and processing of the corresponding CONNECTION_SYNC. The data associated with this reply may include, but is not limited to: connection reference.

CONNECTION_SEND_NOTIFY

A L5NIC acknowledgement of successful transfer of a certain amount of data of an offloaded TCP connection. The data associated with this notification may include, but is not limited to: connection reference, amount of data successfully transferred over the connection (acknowledged by the remote connection peer). Host may use this notification to free the data respective of the connection and queued by the application.

FIGURE 10 B

CONNECTION_RECIEVE_NOTIFY

Accelerator notifies reception of additional data on the connection into anonymous or connection specific host buffers. The data associated with this notify maybe, but not limited to: connection reference, some buffer identification and amount of data posted into the buffer.

LINK_LAYER_RECEIVE_NOTIFY

L5NIC notification of reception of additional Layer 2 into host buffers. The data associated with this notification may include, but is not limited to: buffer identification, amount of data posted into the buffer.

CONNECTION_NOTIFY

L5NIC notifies host of a change in a connection state. The data associated with this notification may include, but is not limited to: connection reference, notification type (connection is ESTABLISHED, disconnected, timed-out, or gracefully closed), connection state, etc.

CONNECTION_ROUTE_NEEDED_NOTIFY

L5NIC notifies host when a new route is needed for the connection. The data associated with this notification may include, but is not limited to: connection reference.

ASYNCH_BUFFERS

A command sent in order to post anonymous receive buffers to the L5NIC. These buffers are used for the received TCP data or Layer 2 data, which is not a data directly associated with the offloaded connections (i.e., all TCP traffic associated with the offloaded connections is processed on the L5NIC). The data associated with this command, in a case of TCP/IP connection, may include, but is not limited to: list of buffers in Host memory, buffer lengths.

LINK_LAYER_SEND

A request sent in order to send Link Layer packet via one of the L5NICs. The data associated with this command, in a case of TCP/IP connection, may include, but is not limited to: packet length, list of buffers in a host memory with the data, which comprises the packet and buffer length.

LINK_LAYER_SEND_NOTIFY

L5NIC acknowledges successful transfer of certain number of Link Layer packets. The data associated with this notification may include, but is not limited to: number of transmitted packets. Host may use this notification to free the data associated with the transmitted packets.

FIGURE 10 C

SET_PARAMETERS

A command sent in order to update global (non-connection specific) TCP/IP parameter of the TCP/IP stack in the L5NIC. The data associated with this command, in a case of TCP/IP connection, may include, but is not limited to: list of parameters to change.

GET_PARAMETERS

This message is sent in order to receive a value of the global non-connection specific TCP/IP parameter from the TCP/IP stack in the L5NIC. The data associated with this command in a case of TCP/IP connection maybe, but not limited to: list of parameters to report.

GET_PARAMETERS_REPLY

L5NIC sends this message in order to respond to the host values requested in a respective GET_PARAMETERS request. The data associated with this response may include, but is not limited to: set of reported values.

GET_STATISTICS

A message sent in order to receive Link Layer, TCP/IP and additional statistics from the TCP/IP stack of the L5NIC.

GET_STATISTICS_REPLY

L5NIC sends this message in response to a respective host GET_STATISTICS request. The data associated with this response may include, but is not limited to: set of reported statistic values.

RESET_STATISTICS

A command sent in order to reset Link Layer, TCP/IP and additional statistics of the L5NIC based TCP/IP stack.

FIGURE 10 D

Group	tl	FreeBSD	Function
Setup	tl open	socket	Creates a transport layer socket
	tl bind	bind	Binds a local address to a socket
Server	tl listen	listen	Listen on a socket
	tl accept	accept	Establish a connection received on a listener
	tl confirm	n/a	ULP confirms the opening of a new connection
Client	tl connect	connect	Starts a connection to the given peer's address
Input	n/a	read	
	tl recv*	recv	Receives data on a connection
Output	n/a	write	
	tl send*	send	Posts data to send over a connection
I/O	n/a	select	
Termination	tl shutdown	shutdown	Shut down the connection
	tl close	close	User request for graceful connection close
	tl disconnect	n/a	User request to tear down the connection
	tl abort	n/a	User request for a connection abort
Administration	tl setsockopt	setsockopt	Management call to set attributes of the connection???
	tl getsockopt	getsockopt	Management call to query attributes of the connection
	tl getsocketname	getsocketname	Get the local address bound to a connection
	tl getpeername()	getpeername()	Get peer's address on a connection
	tl transfer in	n/a	Transfer's a connection into a L5NIC
	tl transfer out	n/a	Transfer's a connection out of a L5NIC
	tl get accelerators()	n/a	Get a list of L5NICs capable of handling a socket
	tl setcbs()	n/a	User set callback routines for events from the TAP
	tl sync	n/a	Synchronization of user application with socket state
Callbacks	tl newconn_cb()	n/a	User callback for request for new connection setup
	tl_transferredout_cb()	n/a	Acknowledges completion of connection transfer out operation
	tl connected_cb()	n/a	Called on completion of an active connection attempt
	tl_disconnected_cb()	n/a	Reports that a connection was asynchronously disconnected (moved to closed state)
	tl_notify_cb()	n/a	User callback for any exception on the connection
	tl_sync_cb()	n/a	Acknowledges completion of sync operation

* - TOE (TCP offload engine) only commands to support generic TCP

FIGURE 11